

Package: mapsf.gui (via r-universe)

June 30, 2026

Title Create Thematic Maps Interactively

Version 0.1.0

Description A 'Shiny' application to create thematic maps interactively, based on the 'mapsf' package. Features include: a user-friendly interface to create and customize thematic maps without coding, support for various map types (choropleth, proportional symbols, etc.) and customization options (colors, legends, etc.), R code generation to ensure reproducibility and export options to save maps in different formats (PNG, SVG).

License GPL (>= 3)

URL <https://codeberg.org/riatelab/mapsf.gui>,
<https://riatelab.r-universe.dev/mapsf.gui>

BugReports <https://codeberg.org/riatelab/mapsf.gui/issues>

Depends R (>= 4.0.0)

Imports base64enc, bslib (>= 0.10.0), colourpicker, Ckmeans.1d.dp, DT, formatR, jpeg, graphics, grDevices, htmltools, mapsf (>= 1.2.1), phosphoricons, png, sf, shiny (>= 1.12.0), shinyjs (>= 2.0.0), shiny.i18n, sortable, stats, utils

Language en-US

Encoding UTF-8

RoxygenNote 8.0.0

Suggests shinytest2, testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Matthieu Viry [cre, aut] (ORCID:
<<https://orcid.org/0000-0002-0693-8556>>), Timothée Giraud [aut]
(ORCID: <<https://orcid.org/0000-0002-1932-3323>>), Hugues Pecout
[aut] (ORCID: <<https://orcid.org/0000-0002-0246-0954>>)

Maintainer Matthieu Viry <matthieu.viry@cnrs.fr>

Config/pak/sysreqs libabsl-dev cmake libgdal-dev gdal-bin libgeos-dev make libicu-dev libjpeg-dev libpng-dev libuv1-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev zlib1g-dev

Repository <https://mthh.r-universe.dev>

Date/Publication 2026-06-29 14:20:02 UTC

RemoteUrl <https://github.com/cran/mapsf.gui>

RemoteRef HEAD

RemoteSha 31f584e3ca4c520a3dc6327fd0cc372e338d98a9

Contents

run_app	2
Index	4

run_app	<i>Run the mapsf.gui Shiny app</i>
---------	------------------------------------

Description

Run the mapsf.gui Shiny app

Usage

```
run_app(data = NULL, viewer = "browser", stop_on_close = TRUE)
```

Arguments

data	Optional list of sf objects to load at launch
viewer	Viewer type: "browser" (default), "pane", or "dialog"
stop_on_close	If TRUE (default), the app will stop when the viewer is closed. Set to FALSE to keep the app running. Note that reloading the app in your browser (with F5 for example) counts as closing the viewer, so if you set this to TRUE, the app will stop when you reload it. Setting this to FALSE allows you to reload the app in your browser without stopping it, but you will need to stop the app manually (by doing Ctrl + C in the console or clicking the stop button in RStudio) when you are done.

Value

This function opens a browser, a pane or a dialog and returns nothing.

Examples

```
if (interactive()) {  
  # Run the app in the browser with no data  
  run_app()  
  
  # Run the app in a dialog with no data  
  run_app(viewer = "dialog")  
  
  # Run the app with example data from mapsf package  
  mtq <- mapsf::mf_get_mtq()  
  run_app(data = list(mtq = mtq))  
  
  # Run the app with multiple example data and without stopping on close  
  # (so that you can reload it in the browser without stopping it)  
  mtq <- mapsf::mf_get_mtq()  
  pts <- mapsf::mf_get_mtq("points")  
  run_app(data = list(mtq = mtq, pts = pts), stop_on_close = FALSE)  
}
```

Index

run_app, [2](#)